Guide pratique

Maintenir et sécuriser son blog sous

Franck Ridel

2018

Sommaire

- Introduction …….page 3

Facile

- Les mises à jour …….page 4

- Vérifiez les vulnérabilités d'une extension ou d'un thème ……page 9

- Cachez le nom et le login des utilisateurs ……..page 12

- Installez des plugins de sécurité …….page 16

- Comment trouver un VRAI mot de passe facile à retenir ……...page 19

Plus difficile

- Comment cacher le numéro de version de Wordpress …….page 24

- Comment fermer les ports inutiles du serveur …….page 26

Bonnes pratiques

- Faîtes une veille régulière sur les l'actualité de Wordpress …….page 30

- Faîtes régulièrement des backups …….page 31

- Bannissez le FTP au profit du SFTP ……..page 32

- Privlilégiez le serveur dédié ou le VPS au serveur mutualisé ……..page 33

- Conclusion ……..page 35

- Glossaire …….page 36

Introduction

Si vous lisez ce guide, c'est probablement parce-que votre blog ou votre site web est propulsé par **Wordpress**. Depuis quelques années, ce dernier a grossi exponentiellement dans le monde des blogs, à tel point qu'il en est presque devenu synonyme *(**Drupal** et **Joomla** ont quand-même une place importante)*. Quelques rudiments et un bon tutoriel suffisent pour que tout le monde puisse installer un blog **Wordpress** hébergé par **OVH, 1and1 ou Gandi** par exemple, pour ne citer que ces 3 hébergeurs. Mais pour les simples utilisateurs *(C'est à dire les non-barbus qui ne sont ni développeurs, ni administrateurs système)*, la simplicité apparente de la maintenance de **Wordpress** peut se révéler fructueuse pour toute personne malveillante qui, si vous n'êtes pas assez rigoureux, peut compromettre votre site web ou votre blog.

Nous allons voir dans ce guide comment sécuriser facilement **Wordpress** avec quelques notions simples et d'autres un peu plus complexes, mais accessibles à tout le monde. Il faut garder en tête qu'**en sécurité informatique, le risque zéro n'existe pas**. Une chose infaillible aujourd'hui sera surement faillible demain, jusqu'à ce que cette faille soit rectifiée après-demain, le temps qu'une nouvelle faille apparaisse le jour suivant. Mais avec de bonnes pratiques quotidiennes, ou du moins régulières, vous pourrez éviter les vulnerabilités les plus courantes et par conséquent éviter d'être la cible de pirates pré-pubères qui font leurs premiers pas de **crackers**[1].

Un **blog Wordpress bien entretenu**, c'est un blog que vous pourrez **garder longtemps** et qui sera **difficile à compromettre**.

<u>IMPORTANT</u> : Les mots ou acronymes écrit en rouge et marqués d'un astérisque (*) sont expliqués dans le glossaire en page 36.

1. Les médias grand public parlent toujours de "**hackers** *qui ont piraté tel site internet*", sans vraiment utiliser le vocabulaire adapté ni comprendre réellement ce que ça signifie. Ils répandent de fait des préjugés très éronnés à leurs téléspectateurs, auditeurs ou lecteurs qui pensent naïvement que *"Si Morandini l'a dit, c'est forcément vrai"*.

Les **hackers sont à l'origine d'internet**. Ils le construisent et le sécurisent quotidiennement depuis les années 70.
Les **crackers cassent tout ce qu'ils peuvent** sans avoir de réelles connaissances, juste parce-que… Voilà !

La différence entre **hacker** et **cracker** est la même qu'entre un **serrurier** et un **cambrioleur** avec un pied de biche.

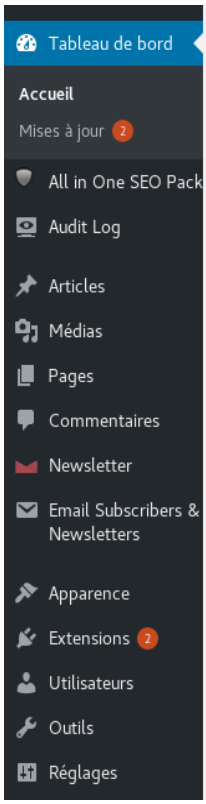
Les mises à jour

Commençons par la partie la plus facile, mais pourtant la plus importante, de ce guide : **Comment** et surtout **pourquoi** mettre à jour **Wordpress**.

Comme dit dans l'introduction, en informatique les failles sont nombreuses, courantes et inévitables. C'est pour cette raison que des **CMS*** *(**C**ontent **M**anagement **S**ystem)* comme **Wordpress**, **Drupal** et **Joomla**, ainsi que des systèmes d'exploitation comme **Windows**, **MacOS**, **GNU/Linux** et la plupart des logiciels qui y sont installés proposent souvent des mises à jour. Parfois, c'est seulement pour avoir de nouvelles fonctionnalités, mais bien souvent ce sont des mises à jour de sécurité ou des corrections de bugs qui peuvent aussi avoir une incidence sur la sécurité. En les mettant régulièrement à jour, vous évitez ainsi à votre site d'être exposé à des vulnérabilités.

II. Mise à jour de Wordpress côté client

Ce qu'on appelle en informatique le "côté client", c'est l'interface avec laquelle les internautes vont interagir sur votre site web ou votre blog pour lire vos derniers articles, laisser un commentaire ou souscrire à vos newsletters. C'est la face visible, contrairement au "côté serveur " du site web ou du blog en question. C'est la partie aussi qui contient beaucoup de plugins utilisables par vos visiteurs *(Formulaires de commentaire, souscription à la newletter, forum, tchat, etc...)* et qui est une cible aussi privilégiée, parce-que facilement accessible par les crackers *(Bien souvent pour trouver des failles **XSS***)*. Nous allons ici mettre à jour le côté client de Wordpress, incluant ses extensions et ses thèmes.



a) Mise à jour des extensions de Wordpress

Dans le menu à gauche, cliquez sur "**Mises à jour**" ou "**Extensions**" quand une notification rouge indiquant le nombre de mises à jour apparait.

Dans la partie “**Extensions**”, en haut ou en bas du tableau affichant les mises à jour disponibles, cochez la case“**Tout sélectionner**”.

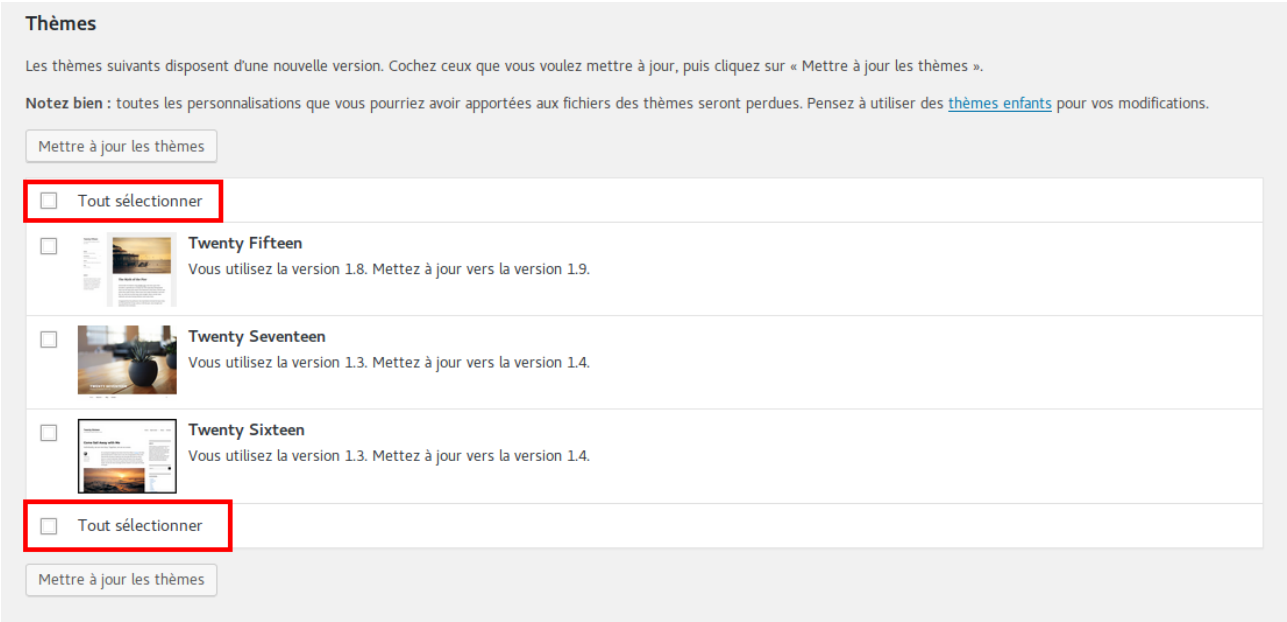
Et enfin cliquez sur un des boutons “**Mettre à jour les extensions**”

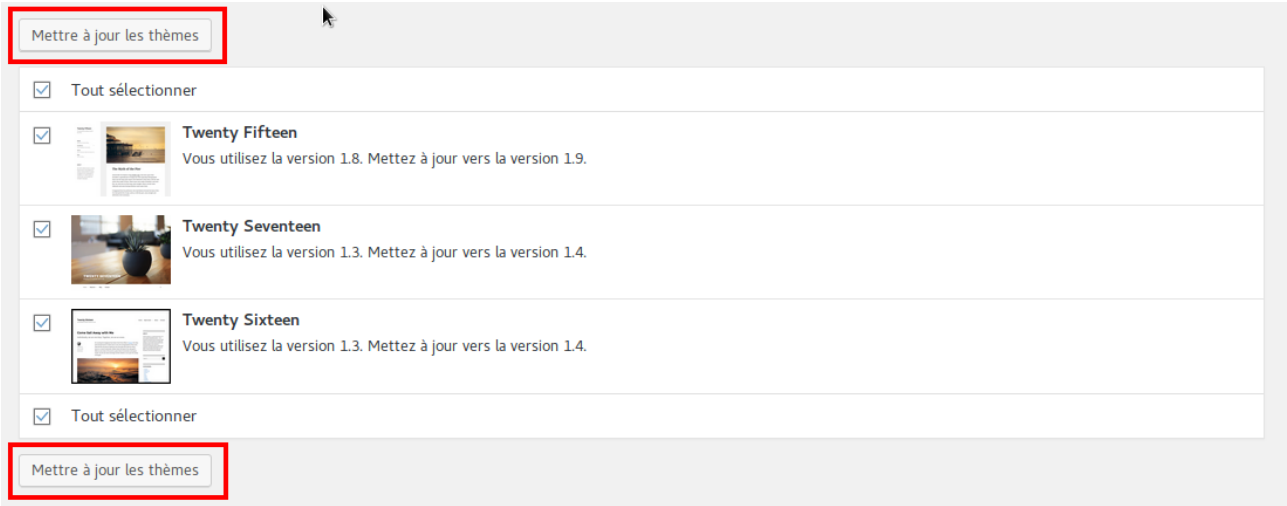
Voilà, patientez quelques secondes… Quand la mise à jour est terminée, une page s’affiche résumant les versions des plugins installées. C’est quelque chose que vous pouvez faire régulièrement et très rapidement, même lorsque vous avez peu de temps libre.

b) Mise à jour des thèmes

Pour mettre à jour les thèmes, qui font aussi partie du côté client, les étapes sont sensiblement les mêmes que pour les extensions. Sous la partie **“Extensions”**

sur la page de mises à jour se trouve une partie **“Thèmes”**. Il vous suffit aussi de cliquer sur **“Tout sélectionner”** et sur **“Mettre à jour les thèmes”**.





c) Mise à jour de la version de Wordpress

Par le passé, il fallait avoir quelques connaissances en **administration système*** et connaître la commande qui va bien pour mettre à jour un **serveur***. Au 21e siècle, à l’époque du **web 2.0***, une souris et quelques clics suffisent. Aujourd’hui vous pouvez même le faire bien souvent, comme pour Wordpress, en quelques secondes à partir de votre smartphone pendant votre pause-déjeuner ou en attendant le métro. Même si le côté serveur est invisible par les internautes lambda, c’est une cible de choix pour les personnes mal intentionnées qui vont essayer de pénétrer votre base de données *(Bien souvent par des **injections SQL*** ou par **bruteforce***, mais pas*

seulement) pour récupérer par exemple les données de vos abonnés ou avoir un accès administrateur sur votre site pour le modifier ou le supprimer.

Pour mettre à jour la version de Wordpress, ou pour vérifier si elle l'est, vous pourrez voir un arvertissement sur le tableau de bord :

Si vous préférez, vous pouvez aussi afficher la même page que pour les extensions *(Onglet "**Mise à jour**" dans le menu en haut à gauche)*.

Quand une mise à jour est disponible, dans la partie “**Mise à jour de Wordpress**” en haut de page, un bouton “**Mettre à jour maintenant**” est visible. Ce bouton vous permet de télécharger la nouvelle version de Wordpress et de l’installer automatiquement.

Quand Wordpress est à jour, vous avez la possibilité de réinstaller la version actuelle, au cas où votre blog aurait subit des modifications non-sollicitées de votre part *(Un cracker qui serait passé par là par exemple)*, ou si vous avez vous-même effectué des modifications dans le code source de Wordpress et que vous vous dîtes *“J’ai tout cassé” (Ce que je désonseille fortement si vous n’avez pas de bases solides en **PHP***)*.

Ces quelques conseils sur les mises à jour sont très faciles à mettre en oeuvre. Peut-être les connaissiez vous déjà avant de lire ce guide, mais le plus important était de mettre en avant le “**pourquoi**” plutôt que le “**comment**”.

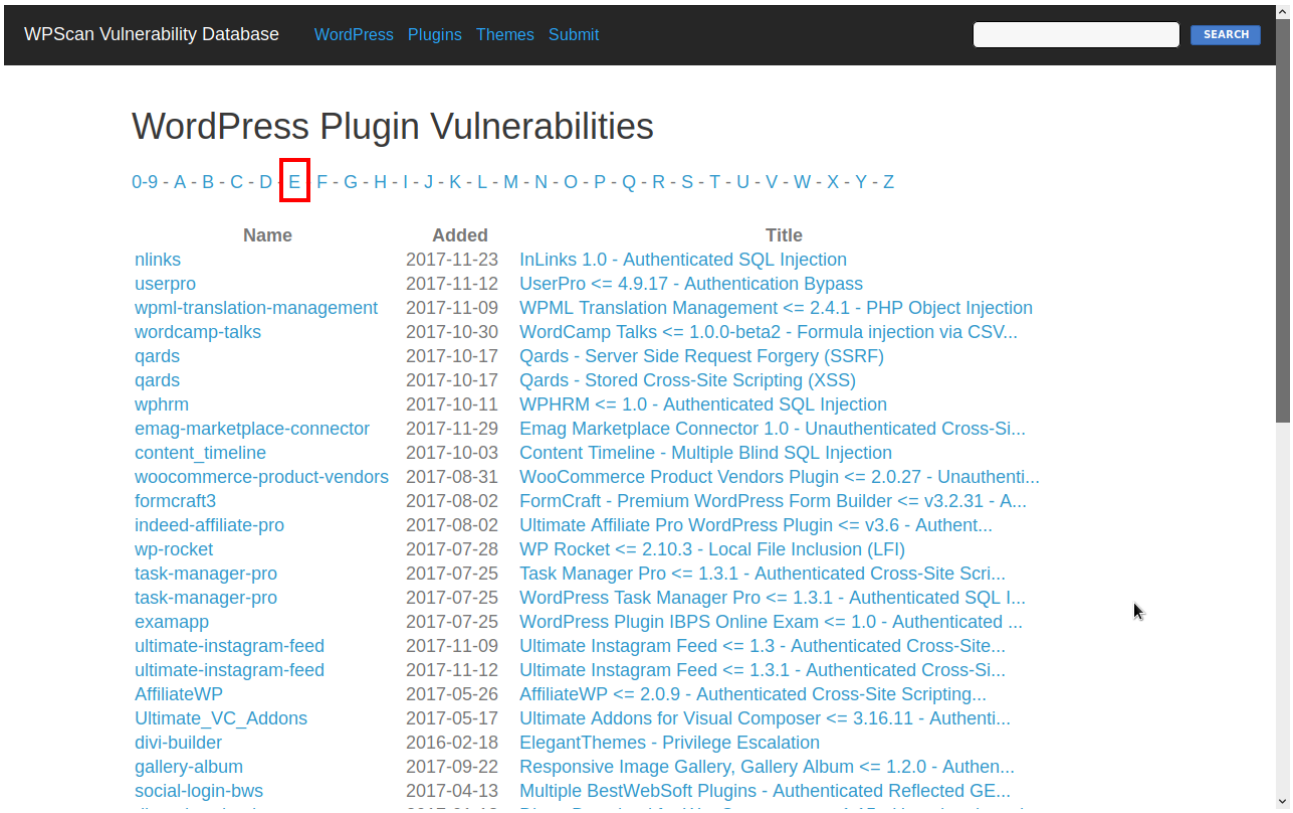
Vérifiez les vulnérabilités d’une extension ou d’un thème

Avant d’installer une extension ou un thème, pensez à **vérifier ses vulnérabilités**. La plupart d’entre eux ont eu ou auront probablement une faille. **En informatique rien est infaillible**, ça peut souvent arriver. Mais si vous voyez que certains en ont régulièrement ou plus souvent que la moyenne, mieux vaut essayer de trouver une alternative. Wordpress étant le CMS le plus répandu, il n’est pas rare de trouver plusieurs extensions qui proposent le même résultat ou des thèmes qui se ressemblent *(Thème que vous pouvez de toute façon personnaliser, mais ce n’est pas l’objet de ce guide)*. Pour vérifier simplement et rapidement les failles passées, vous pouvez consulter le site https://wpvulndb.com qui les répertorie en plusieurs parties. Dans notre cas, seules 2 nous intéressent :

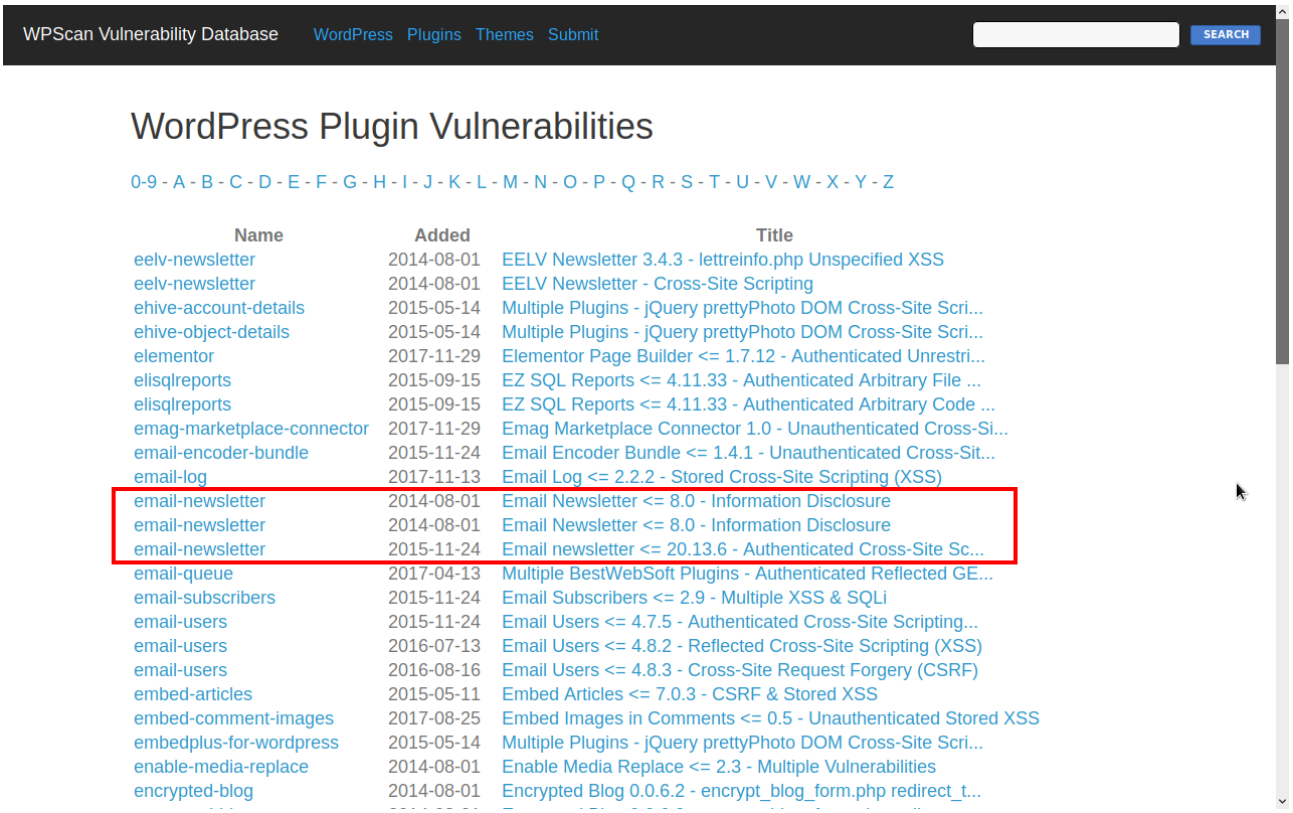
Pour les thèmes :
https://wpvulndb.com/themes

Pour les plugins :
https://wpvulndb.com/plugins

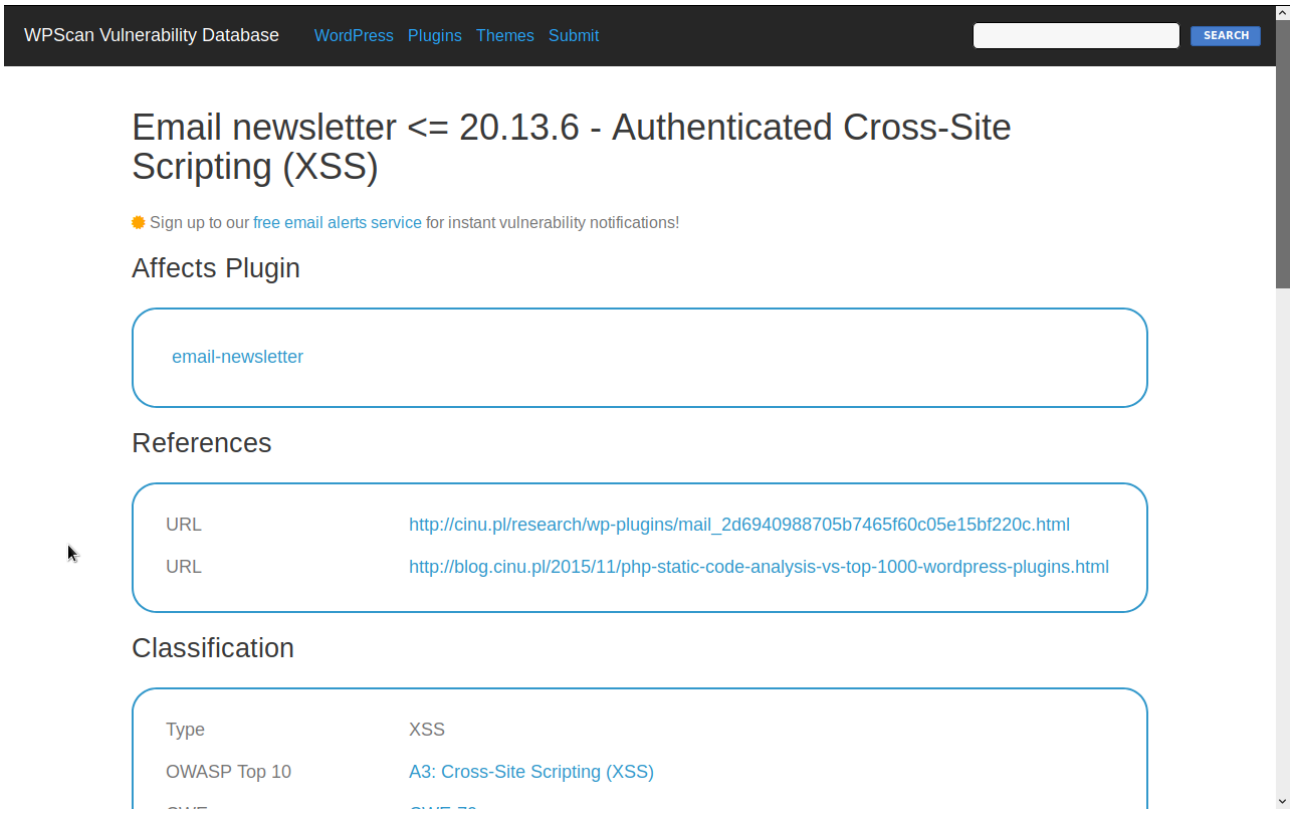
Si on teste par exemple avec l'extension **email newsletter**, vous pouvez taper son nom dans le champs de recherche en haut à droite. Si vous n'êtes pas sûr de l'orthographe, il suffit taper sur l'initiale de l'extension *(Ici la lettre **E**) :*



...et on fait défiler jusqu'à tomber dessus *(À l'heure où j'écris ces lignes il se trouve en page 2)*.



Vous pouvez cliquer sur le nom de chaque lien pour voir la gravité de la faille en question *(Ici une faille XSS)*.



À l'heure actuelle il n'y a eu que 3 failles, ce qui est acceptable. En moyenne la plupart des extensions en ont une seule, parfois deux. À partir de 5 ou 6 on peut remettre en doute sa fiabilité *(Mais à juger au cas par cas comme nous allons le voir au chapitre suivant avec l'extension* ***Stop User Enumeration****)*.

Prenez cette habitude afin de ne pas ajouter de nouvelles vulnérabilités à votre blog ou votre site, et le rendre ainsi plus *"fragile"*. Ça ne prend que quelques minutes et ça peut éviter quelques dégâts.

CACHEZ LE NOM ET LE LOGIN DES UTILISATEURS

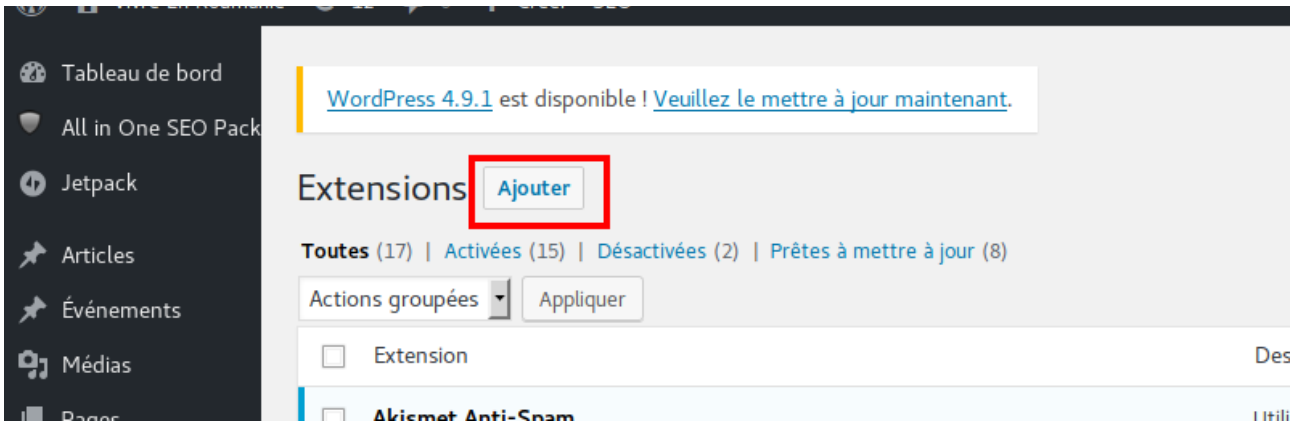
Maintenant que la plupart des “défauts de fabrication” sont écartés, nous allons éliminer les quelques possibilités restantes.

Une des techniques utilisées par les crackers est de **récupérer à l’aide d’un script** *(Comme **Wpscan*** par exemple)* **le nom et le login de tous les utilisateurs**. Ceci afin de pouvoir tester différents mots de passe jusqu’à trouver le bon. De nos jours, c’est possible d’automatiser cette recherche à l’aide d’un script comme **Hydra*** qui parcours un dictionnaire de mot de passe *(C’est possible aussi avec WPScan)* et les teste un par un *(Un dictionnaire comme **RockYou** contient **32 millions de mot de passe**, mais on peut aussi en générer soi-même avec un script comme **Crunch** en lui indiquant certains paramètres comme les caractères utilisés ou la longueur du mot de passe)*.

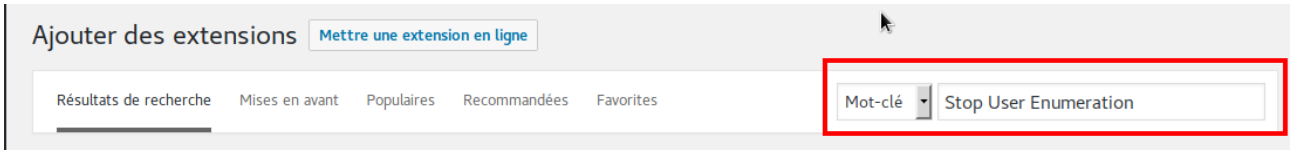
Pour éviter la récupération du couple login/nom d’utilisateur, plusieurs méthodes sont possibles, Nous allons utiliser ici la plus simple et installer une extension prévue pour cette tâche : **Stop User Enumeration**.

Si vous avez bien suivi l’étape précédente, vous avez surement déjà affiché l’historique de ses vulnérabilités passées. À l’heure où j’écris ces lignes il y en a eu **5** au total : **une en 2014**, **une en 2016** et **3 en 2017**. La plupart sont des failles qui permettent de bypasser *(De contourner)* l’extension. Si on pèse le pour et le contre, on peut se dire que ce bypass n’ajoute pas de vulnérabilité supplémentaire à votre site mais rend seulement l’extension inutile pour tout cracker qui analyse votre site. D’autres méthodes sont disponibles pour cacher les noms et logins utilisateurs, mais elles sont un peu plus complexes et nécessitent d’agir directement sur la base de données avec **PhpMyAdmin*** ou **MySQL*.** De fait, on va quand-même installer **Stop User Enumeration**. Ce guide s’adressant aux utilisateurs pas ou peu expérimentés, nous allons voir la méthode la plus simple concernant ce point.

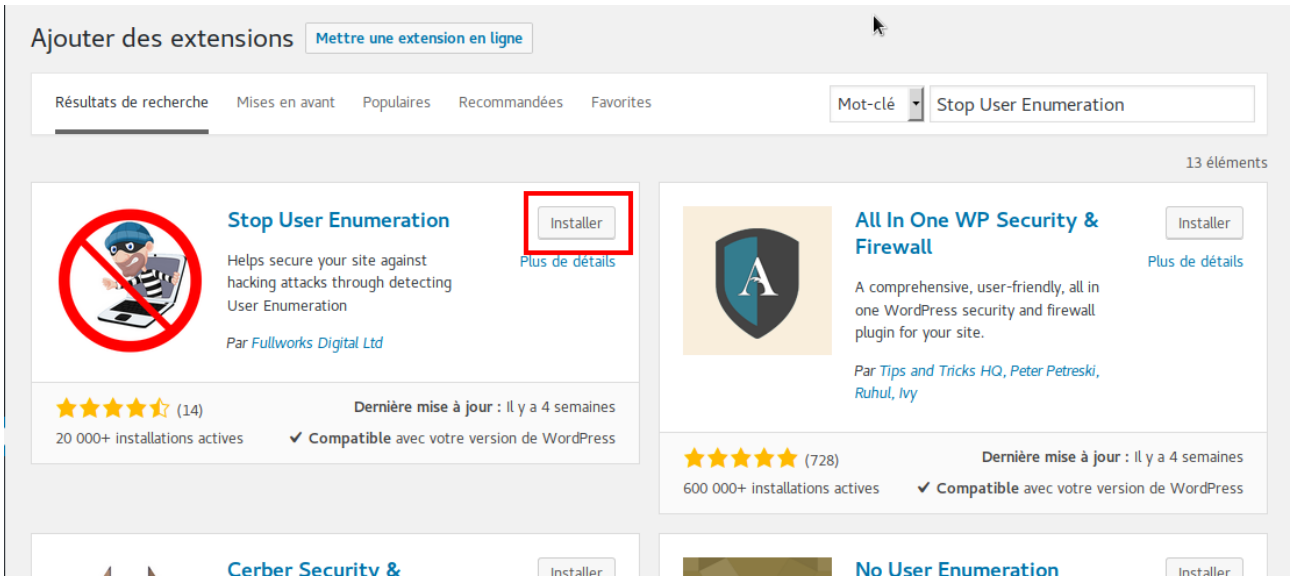
Pour ce faire, si vous n'avez encore jamais installé d'extension, nous allons voir rapidement comment procéder. Dans le menu à gauche, cliquez sur l'onglet "**Extensions**" dans le menu à gauche. Vérifiez dans la liste des extensions qui s'affiche si l'extension **Stop User Enumeration** est déjà installée *(Vous pourriez avoir oublié ou un ami connaisseur aurait pu le faire pour vous)*. Si ce n'est pas le cas, cliquez en haut de la page sur le bouton "**Ajouter**".

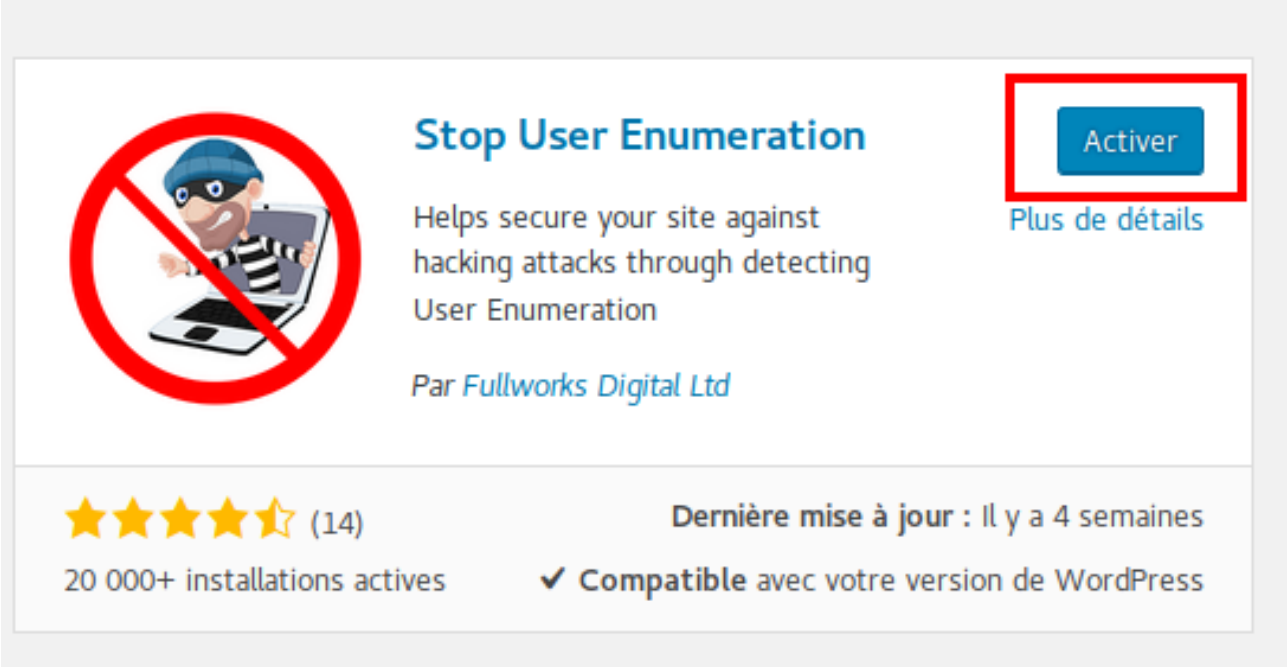


Sur la page qui vient d'apparaître, cliquez dans le champ de recherche en haut à droite et tapez "**Stop User Enumeration**", la page se rafraîchira automatiquement.



Cliquez sur “**Installer**”, et ensuite sur **“Activer”**, quelques secondes suffisent pour effectuer la tâche. Votre blog refusera désormais de fournir nom d’utilisateur et login à toute personne utilisant Wpscan pour les récupérer.





Ceci étant fait, il faut tout de même garder en tête qu'il est possible de les deviner. Si vous répondez aux commentaires de vos utilisateurs ou s'ils recoivent une newsletter à chaque nouvel article, votre nom d'utilisateur est probablement affiché. C'est pour cette raison qu'il vous est conseillé d'utiliser un compte "***administrateur***" qui sert **uniquement à administrer le site** et un compte "***auteur***" pour **écrire les articles et répondre aux commentaires**. Si ce dernier compte devait être compromis, le pirate sera un peu plus limité dans ses actions et surtout ne pourra pas modifier votre site ou effectuer des tâches réservées à l'administrateur.

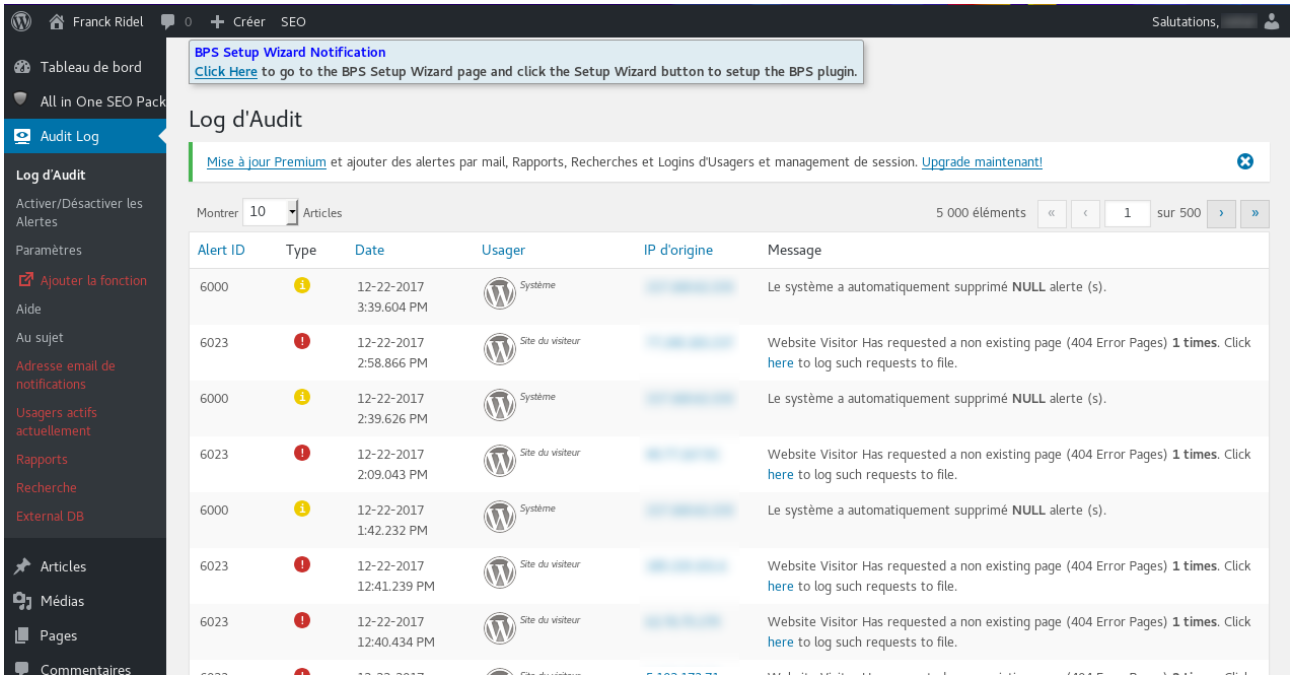
Il est bien évidemment déconseillé pour le pseudo du compte administrateur d'utiliser les habituels "**admin**", "**administrateur**", "**root**", "**webmaster**", "**webmestre**" et autres noms du même genre qui seront les premiers à être testés par une personne voulant obtenir les droits administrateurs de votre site.

Il est important aussi de faire des **backups*** *(Des sauvegardes)* afin de pouvoir restaurer le site s'il venait à être modifié à votre insu *(Plus de détails à la **page 31**)*.

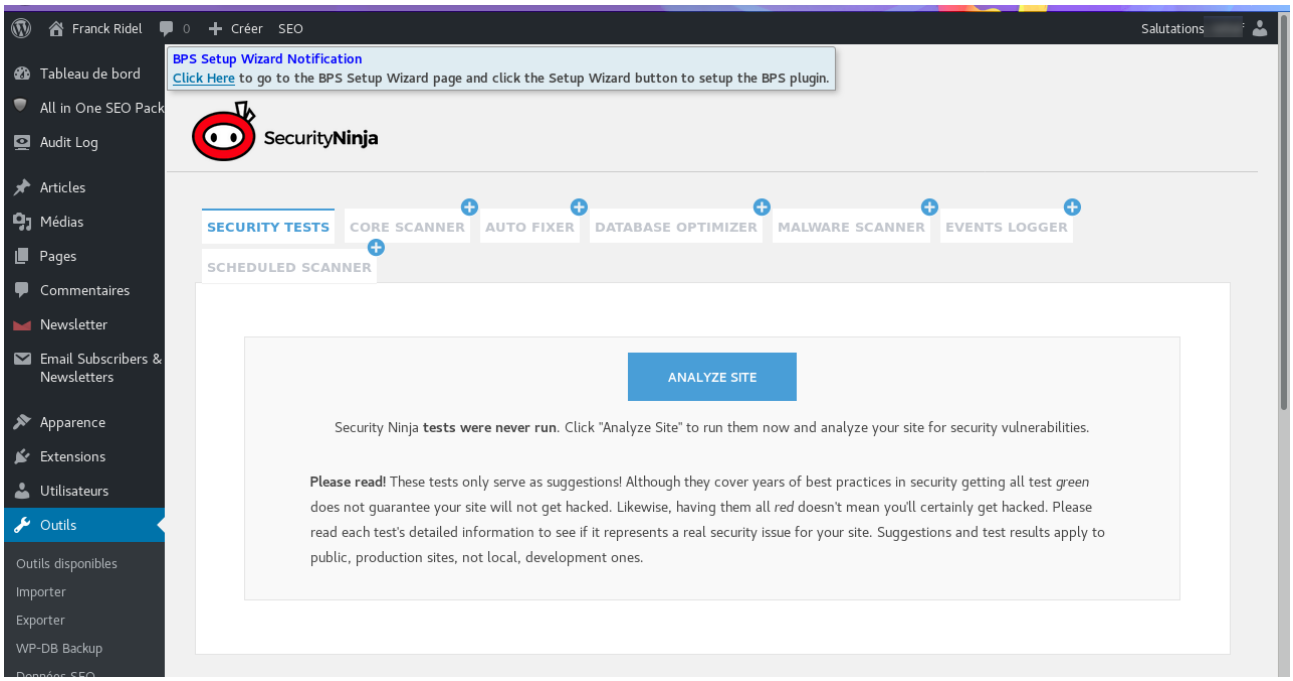
INSTALLEZ DES PLUGINS DE SÉCURITÉ

Il existe de nombreuses extensions vous permettant de sécuriser votre blog. Certaines sont gratuites, d'autres payantes. Il est bien évidemment impossible de toutes les réportorier ici, nous allons seulement en voir quelques unes parmi les plus efficaces. À vous ensuite d'effectuer quelques recherches en fonction de vos besoins.

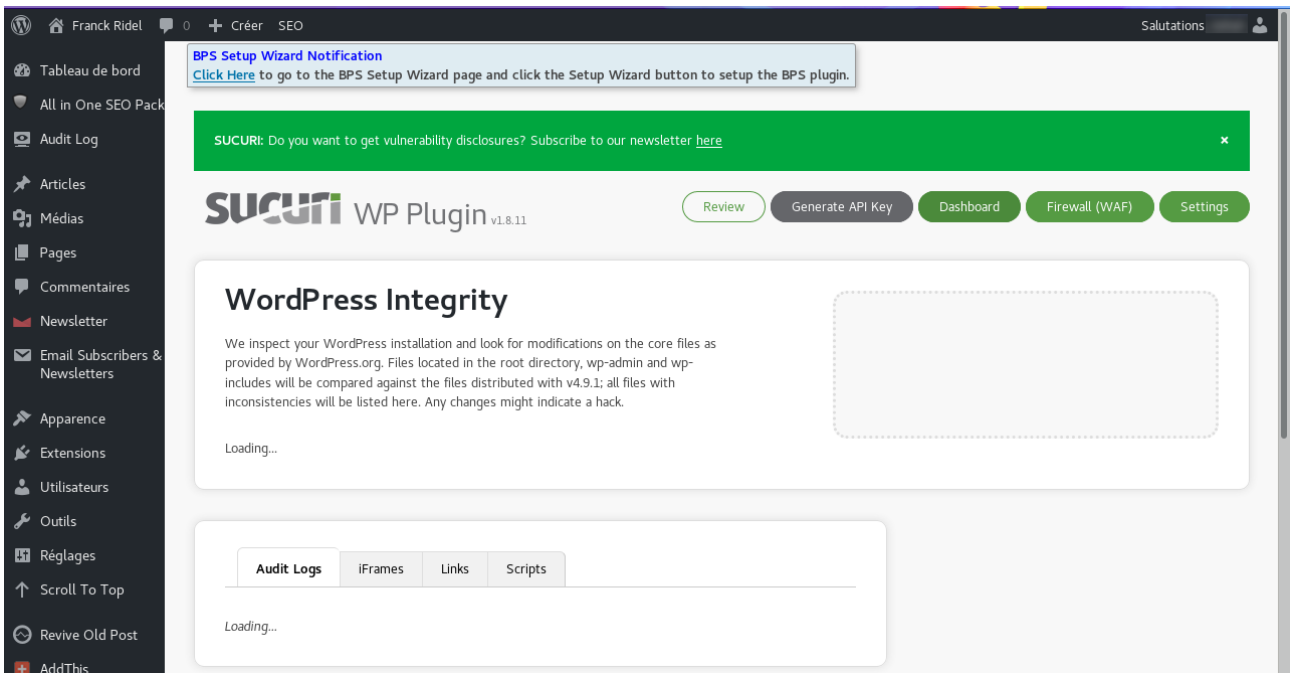
WP Security Audit Log : Cette extension vous permet de lister les logs *(Les connexions) sur* votre site afin de savoir quels sont les utilisateurs qui se sont connectés, lesquels ont été créés ou supprimés, ceux qui ont uploadé quelque chose, etc… Un niveau de confiance ou des alertes sont assignées à chaque log afin de vous permettre de faire le tri plus rapidement.



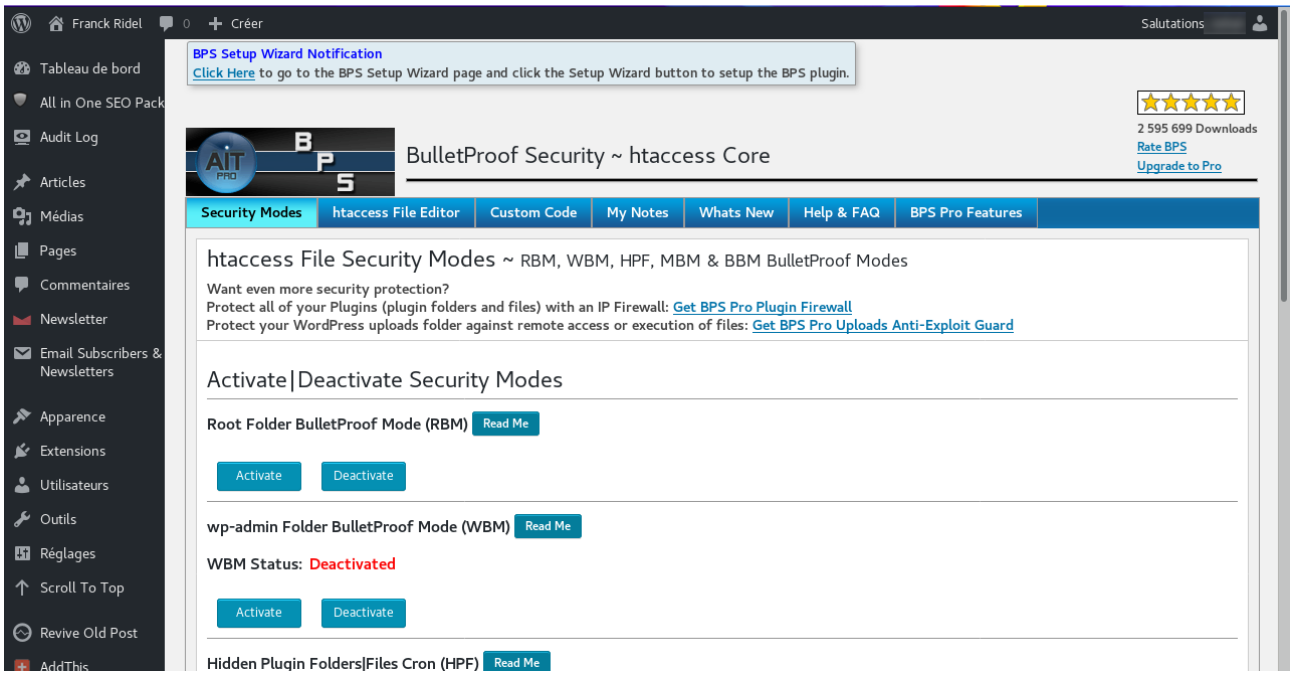
Security Ninja : Cette extension vous permettra d'automatiser beaucoup de tests de vulnérabilités, aussi bien des attaques bruteforce que des attaques **0-day***, ainsi que Apache, PHP, les attaques par brute-force ou la base de données.



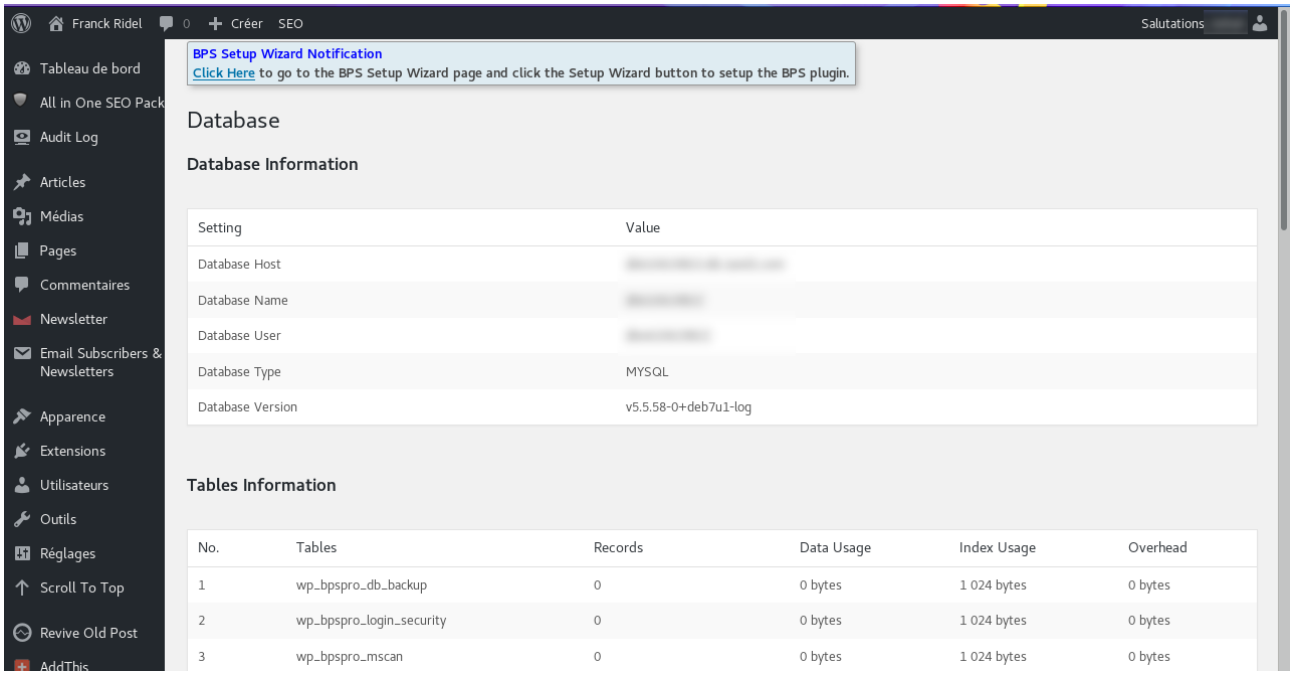
Sucuri Security : Cette extension permet d'ajouter une couche de sécurité supplémentaire en analysant les fichiers modifiés sur votre site, leur intégrité et en recherchant des vulnérabilités afin de renforcer sa sécurité.



Bulletproof Security : Cette extension ajoute aussi une sécurité supplémentaire en ajoutant un firewall *(**pare-feu***)* à votre site et en protégeant le **.htaccess***, les connections à votre base de données, en proposant de faire des backups de cette dernière, ainsi que la vérification des **cookies***.



WP-DBManager : Pour terminer cette section, cette extension vous permettra de pousser un peu plus loin les backups proposés par l'extension ci-dessus, avec beaucoup d'options de sauvegarde ainsi que de déplacements de certains fichiers.



COMMENT TROUVER UN VRAI MOT DE PASSE FACILE À RETENIR

L'étape précédente nous permet de mettre quelques bâtons dans les roues des crackers. Mais quelques petits bâtonnets seulement. Même en bloquant la récupération des données des utilisateurs, les noms et logins peuvent être devinés. Nous allons donc maintenant mettre un gros tronc d'arbre en travers de leur route et créer un **VRAI** mot de passe.

Si j'insiste sur le mot "**VRAI**", c'est parce-qu'il va falloir éradiquer définitivement une chose de votre esprit : " **123456** " n'est pas un mot de passe, pas plus que "**motdepasse**" ou "**azerty**", ni "**patricksebastien**" ou "**reblochon74**". Beaucoup de comptes, pas seulement **Wordpress** mais sur les réseaux sociaux, les forums et mêmes des adresses email, sont compromis à cause de cette très mauvaise pratique qui existe encore aujourd'hui. Oui oui, des gens utilisent vraiment ce genre de mot de passe. Et en plus ils utilisent généralement le même sur tous leur comptes. Je vous laisse imaginer la joie d'un cracker boutonneux qui récupère quelques profils d'une personne qui met " **123456** " partout.

Les personnes malveillantes utilisent aujourd'hui, comme dit dans le chapitre précédent, des logiciels avec des dictionnaires de mots de passe. Même si leur contenu est souvent en anglais, il en existe qui sont adaptés à une langue bien spécifique, et le français étant une des langues les plus parlées au monde, nous avons autant à craindre que les internautes anglophones ou presque. Les mots, noms ou prénoms trop courants seront trouvés en quelques secondes. Les combinaisons de type *"prénom + département" ou "prénom du fils+ année de naissance"* comme " ***nadine75*** " ou " **matéo2012** " ne sont pas plus sécurisées. La plupart des scripts utilisant des dictionnaires de mots de passe permettent aussi de tester des préfixes et des suffixes. Si *"nadine"* par exemple se trouve dans le dictionnaire utilisé par le pirate, le script va tester différents suffixes *("**nadine1**", "**nadine2**", "**nadine3**", etc… puis "**nadine01**", "**nadine02**", "**nadine03**", etc… puis avec des majuscules "**Nadine01**", "**Nadine02**", "**Nadine03**", etc… ou même chose avec des préfixes...)*.

Ce genre de script peut tester plusieurs miliers mots de passe ou de combinaisons par seconde, ça peut aller très vite.

C'est pour cette raison que beaucoup de personnes se font subtiliser leurs données et/ou leurs comptes en rejetant la faute sur les *"vilains pirates"*. Je pense a contrario que le tort est souvent partagé, dans bien des cas...

Il existe différentes manières d'utiliser des mots de passe complexes. Une des plus utilisées à l'heure actuelle est d'utiliser un gestionnaire de mots de passe. Vous n'en avez qu'un seul à en mémoriser et le logiciel se charge de retenir tous ceux que vous utilisez. Il peut aussi en générer aléatoirement en choisissant la longueur et les caractères utilisés. L'inconvénient de cette méthode, bien qu'elle soit vraiment très efficace, est que si vous empruntez un PC chez un ami pour vérifier vos derniers commentaires ou vos mises à jour, vous devez absolument avoir la "mini-base de données" avec vous, sur une carte SD ou une clé USB. Pas toujours pratique. De plus, les dernières vulnérabilités matérielles dévoilées en janvier 2018 et qui concernent la plupart des processeurs *(Failles connues sous les noms de **Meltdown et Spectre***)* rendent ces gestionnaires moins confidentiels.

Nous allons donc prévoir dans la partie suivante ces moments où vous êtes en parti week-end familial ou en vacances loin de chez vous. Et ce sans votre matériel habituel à cause de la belle-mère qui prend toute la place sur la banquette arrière avec ses 3 valises et la caisse du chat. Plusieurs techniques sont souvent recommendées, nous allons en voir 4 d'entre elles que nous allons combiner progressivement.

I. Une phrase facile à retenir

Une des techniques recommendées est le fait d'écrire une phrase longue *(Pas trop non plus)* mais facile à retenir. Prenons par exemple celle-ci :

" **la vie est un long fleuve tranquille** "

Si nous collons tous les mots cela nous donne :

" lavieestunlongfleuvetranquille "

II. Mélanger majuscule est minuscules

Pour cette partie, vous pouvez faire comme bon vous semble. Mettre une lettre sur deux en majuscule et le reste en minuscule peut être une bonne méthode. Pour la

phrase de la partie précédente, vous pouvez tout simplement mettre la première lettre de chaque mot en majuscule, le reste en minuscule :

“ **lavieestunlongfleuvetranquille** ” devient :

“ **LaVieEstUnLongFleuveTranquille** ”

C’est déjà pas mal. Les titres de films français étant souvent des noms à rallonge *(“De battre mon coeur s’est arrêté”, “La vie dissolue de Gérard Floque”, ou même “Autant en emporte le vent” si vous préférez utiliser un classique de la littérature française)*, vous n’avez que l’embarras du choix. Vous pouvez prendre une réplique célèbre *(“On en as gros, Sir !”)*, un court proverbe *(“Aide toi et le ciel t’aidera”)*, une expression *(“Les routiers sont sympas”)*, bref, quelque chose qui comporte plusieurs mots, qui a un sens, et que vous êtes sûr de retenir.

III. Le Leet Speak

Le Leet Speak est une méthode qui consiste à complexifier un nom, un pseudo ou un texte en remplaçant certaines lettres par d’autres caractères qui y ressemblent visuellement ou qui ressemblent à la même lettre inversée. Poussé à l’extrême, ce procédé peut vite devenir très complexe pour toute personne qui n’est pas habituée à l’utiliser. Mais nous allons commencer très simplement en remplaçant certaines lettres par des chiffres.

Nous allons remplacer :

- Les **A** par des **4**

- Les **E** par des **3**

- Les **I** par des **1**

- Les **O** *(La lettre)* par des **0** *(Zéro)*

- Les **S** par des **5**

Reprenons la phrase de l’étape précédente en appliquant cette méthode :

“ **LaVieEstUnLongFleuveTranquille** ” devient :

“ **L4V1335tUnL0ngFl3uv3Tr4nqu1ll3** ”

Avec ça, nous pourrions nous arrêter là et utiliser ce mot de passe qui est déjà bien complexe. Mais dans les bonnes pratiques les mots de passe doivent aussi contenir des caractères spéciaux.

IV. Utiliser des caractères spéciaux

Sont considérés comme caractères spéciaux les caractères qui ne sont ni des chiffres, ni des lettres. Bien souvent les signes de ponctuation *(! ? , . ; : “ ‘)* mais aussi d’autres symboles *($ € [ ] { }( ) / \ | * % # @)*.

Vous pouvez choisir d’en placer au hasard dans votre mot de passe. Mais ce chapitre étant destiné à créer des mots de passe faciles à retenir, nous allons les insérer en appliquant la méthode du Leet Speak vu precédemment.

Nous allons remplacer :

- Les **C** par des **(** *(Des parenthèses ouvrantes)*

- Les **L** par des **|** *(Des pipes :* ***AltGr + 6*** *sur un clavier AZERTY,* ***au-dessus de l’antislash “\”*** *sur un QWERTY)*

- Les **S** par des **$**

Nous pourrions aller beaucoup plus loin, mais ce sera suffisant. Reprenons notre phrase de la partie précédente :

“ **L4V1335tUnL0ngFl3uv3Tr4nqu1ll3** ”

- Dans les deux premiers mots, “ **La vie** ”, nous n’allons rien changer et garder “ **L4V13** ”

- Dans le troisième mot, “ est ”, nous avons remplacé le “ **S** ” par un “ **5** ” dans la partie 3 de ce chapitre. Mais nous allons maintenant le remplacer par un **$**. “**35t**” devient donc “ **3$t** ”

- Les deux mots “ **un long** ” peuvent rester tels quels, c’est à dire “ **UnL0ng** ”

- Dans le mot “ **fleuve** ” nous pouvons remplacer le “**l**” par un **|** *(pipe)*, ce qui devient “ **F|3uv3** ”

- Enfin, “ **tranquille** ” peut rester aussi comme il est : “ **Tr4nqu1ll3** ”

N’ayant pas de lettre “ **C** ” dans notre exemple, nous n’avons pas eu la possibilité d’utiliser une parenthèse ouvrante, mais vous en aurez probablement l’occasion. Nous pouvons ajouter tout de même un point d’exclamation à la fin de notre phrase.

Si nous résumons ces quelques changements, le mot de passe de la partie précédente “ **L4V1335tUnL0ngFl3uv3Tr4nqu1ll3** ” + un “ **!** ” devient maintenant “ **L4V133$tUnL0ngF|3uv3Tr4nqu1ll3!** ”.

Si cette quatrième partie vous parait un peu compliquée ou obscure, vous pouvez arrêter le procédé à la partie 3, ce qui donnera déjà bien du fil à retordre aux petits crackers qui auront des cheveux blancs plus tôt que prévu, s’ils s’acharnent sur votre mot de passe.

Pour s’assurer que vous ayez bien compris, voici quelques exemples dont vous pourrez vous inspirer *(Directement de la phrase originale jusqu’au résultat final cette fois) :*

- “ **Le gras, c’est la vie !** ” devient “ **L3Gr4$,(‘35tL4V13!** ”

- “ **Les routiers sont sympas !** ” devient “ **L3$R0ut13r5$0nt5ymp4$!** ”

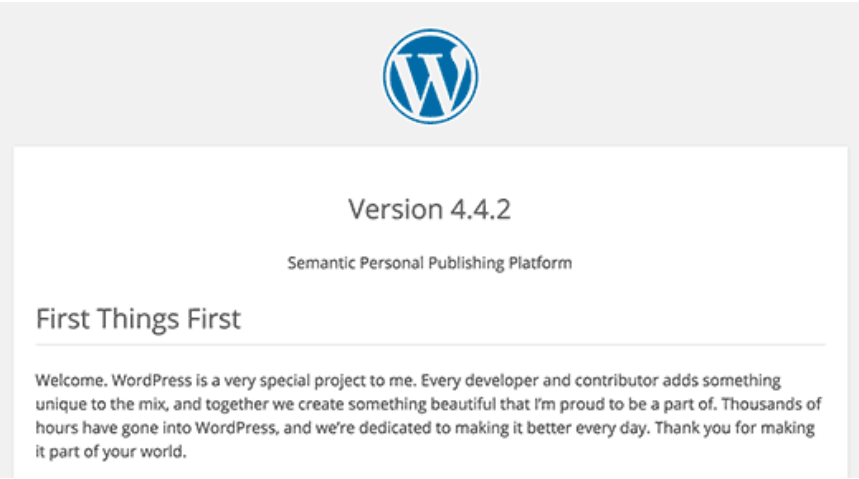
- “ **Je mangerais bien une glace !** ” devient “ **J3M4ng3r41$B13nUn3Gl4(3!** ”

COMMENT CACHER LE NUMÉRO DE VERSION DE WORDPRESS

Le numéro de version de Wordpress peut paraître anodin concernant la sécurité de votre blog. Chacune de ces versions ayant ses propres vulnérabilités, un cracker connaissant la version utilisée sur votre site pourrait cibler les potentielles failles *(En consultant par exemple https://exploitdb-com)* et ainsi gagner du temps. Nous allons donc voir comment cacher cette information.

Remarque : Les étapes qui vont suivre s'adressent aux personnes ayant déjà utilisé un **éditeur de texte** *(**Notepad**++, **SublimeText**, **Gedit**, etc…)* ou un **IDE** *(**Eclipse**, **Netbeans**, **PhpStorm**, etc…), ainsi qu'un **client*** **FTP** (**Filezilla**, **Cyberduck**, etc...) et ayant quelques notions en **PHP**. Si ce n'est pas le cas, mieux vaut demander de l'aide auprès d'une autre personne.

Tout d'abord, sur les anciennes versions de Wordpress, le fichier **readme.html** qui se trouve à la racine de votre site donnait le numéro de version installée. Si pour quelque raison que ce soit vous avez une de ces vieilles versions sur votre site *(Du point de vue sécurité c'est vraiment a éviter)*, vous devrez alors l'effacer. Ce fichier permettait très facilement d'afficher la version de votre site en tapant simplement ***exemple.com/readme.html*** comme vous pouvez le voir ci-dessous :



Ensuite, nous allons modifier le code source d'un fichier du thème que vous utilisez. Les numéros de version peuvent apparaître à différents endroits : Dans le **header**, à la fin des **appels de scripts CSS** et **Javascript** *(Qui sont générés par certaines extensions)* et dans le flux **RSS**.

Nous allons donc ajouter quelques fonctions au fichier **functions.php** de votre thème. Le but ici n'étant pas de faire un cours de programmation mais seulement de sécuriser votre site, je ne rentrerai pas dans les détails du code.

```
function remove_version_generator() { return '';}
add_filter('the_generator', 'remove_version_generator');

function removeVersionCallback($matches) {
return "ver=".md5(print_r($matches, true)."");
}
function removeVersion($url) {
return preg_replace_callback("/ver=[^&]*/", removeVersionCallback, $url);
}
add_filter( 'style_loader_src', 'removeVersion');
add_filter( 'script_loader_src', 'removeVersion')
```

Une fois le code ajouté, sauvegardez et quittez votre **éditeur de texte** ou votre **IDE**. Vous avez terminé.

COMMENT FERMER LES PORTS INUTILES DU SERVEUR

Pour communiquer entre eux, les ordinateurs utilisent différents protocoles *(**HTML** pour les pages web, **SMTP** pour les emails, **FTP** pour les serveurs de fichiers, etc…)*. Parfois, certains protocoles sont ouverts automatiquement par votre système d'exploitation, parfois ils sont ouverts par vous mais vous ne vous en servez pas souvent, voire jamais. Plus le nombre de protocoles ouverts est important, plus vous offrez de possibilités aux crackers de pénétrer votre système, et donc de potentiellement leur donner un accès aux données de votre site. Nous allons donc voir, dans les grandes lignes, comment procéder pour minimiser le nombre de failles. Il existe différentes familles de systèmes d'exploitations (Environ **2/3 des serveurs** de type **UNIX**, incluant les systèmes **GNU/Linux**, dont il existe de nombreuses variantes comme **Ubuntu**, **Debian**, **Red Hat**, **CentOS**, **Fedora**, etc… et les systèmes de type **BSD** comme **FreeBSD**, **OpenBSD**, **NetBSD**. Les autres serveurs utilisent **Windows Server** ou des systèmes d'exploitation comme **Oracle** ou **Solaris** *(Qui sont surtout axés "Entreprise")*. Je donnerai donc seulement quelques indications globales que vous devrez ensuite adapter à votre propre système).

Remarque : Pour cette partie, il est recommandé d'avoir **un minimum de connaissances en réseaux informatiques**. Ces quelques connaissances étant très légères, toute personne sachant se connecter au serveur de son site peut effectuer cette partie et combler ses lacunes avec **une simple recherche Google**. Sinon, vous pourrez demander à votre administrateur ou à une tierce personne de le faire.

Pour laisser uniquement les ports utiles ouverts, nous pouvons procéder de 2 manières :

- Scanner votre serveur pour lister les ports ouverts, et fermer un à un ceux qui sont inutiles.
- Fermer tous les ports et ouvrir un à un ceux dont vous avez besoin.

Comme dis plus haut, la plupart des serveurs utilisent des systèmes **Unix**. C'est donc sur cette famille de système d'exploitation que nous allons nous baser ici.

Méthode 1 : scan réseau et fermeture de ports

Pour scanner votre machine, il existe de nombreuses commandes. La plus répandue est aujourd'hui probablement **nmap**. Elle est généralement disponible sur les distributions **Linux** avec un simple **apt-get install nmap** pour Debian ou **dnf install nmap** pour Fedora ou **yum install nmap** pour Red Hat ou **pkg install nmap** pour FreeBSD, etc... Une fois installé, vous pouvez faire un scan de votre réseau en tapant cette commande :

sudo nmap -PS <URL de votre site ou adresse IP de votre serveur>

Vous obtiendrez alors le résultat assez similaire à ceci :

```
franck  ~  sudo nmap -PS franck-ridel.fr

Starting Nmap 7.60 ( https://nmap.org ) at 2018-01-09 22:03 EET
Nmap scan report for franck-ridel.fr (217.160.0.252)
Host is up (0.050s latency).
Other addresses for franck-ridel.fr (not scanned): 2001:8d8:100f:f000::2e3
rDNS record for 217.160.0.252: 217-160-0-252.elastic-ssl.ui-r.com
Not shown: 942 filtered ports, 55 closed ports
PORT    STATE SERVICE
80/tcp  open  http
81/tcp  open  hosts2-ns
443/tcp open  https

Nmap done: 1 IP address (1 host up) scanned in 316.80 seconds
franck  ~
```

Ensuite, vous pourrez fermer les ports qui vous sont inutiles avec **iptables** *(Installé par défaut sur la plupart des distributions **Linux**)*. Voici la manière de bloquer un port avec cette commande :

iptables -A INPUT -p { NOM_DE_PROTOCOLE } --destination-port {NUMERO_DE_PORT} -j DROP

Vous devrez adapter **NOM_DE_PROTOCOLE** et **NUMERO_DE_PORT**. Un exemple pour bloquer le port FTP *(Pour les serveurs de fichiers)* :

iptables -A INPUT -p ftp --destination-port 21 -j DROP

Vous pourrez trouver sur de nombreux sites des infos sur les ports correspondant aux protocoles qui seront affichés par nmap. Certaines commandes comme **whatportis** peuvent aussi être installées sur vos machines.

```
root@6933e02fba9e:/# whatportis 21
+------+------+----------+----------------------------------+
| Name | Port | Protocol | Description                      |
+------+------+----------+----------------------------------+
| ftp  |  21  |   tcp    | File Transfer Protocol [Control] |
| ftp  |  21  |   udp    | File Transfer Protocol [Control] |
| ftp  |  21  |   sctp   | FTP                              |
+------+------+----------+----------------------------------+
root@6933e02fba9e:/#
```

Méthode 2 : Bloquer tous les ports et ouvrir seulement les ports utiles

Pour cette 2e méthode, vous devrez d'abord identifier de quels ports vous avez besoin. Au moins le **port 80** pour le **HTTP**, probablement le **port 443** pour le **HTTPS**, peut-être d'autres comme le **port 21** pour le **FTP**, etc… Après avoir terminé cette liste, il suffira de fermer la totalité des ports.

ATTENTION : Pour effectuer cette 2e méthode, il faudra bien entendu **avoir un accès direct au serveur.** Si vous êtes connectés en SSH et que vous fermez les ports, vous vous interdirez l'accès au serveur.

Nous allons commencer par effacer les règles existantes de **iptables** :

iptables -t filter -F

iptables -t filter -X

Ensuite, nous allons bloquer la totalité du traffic, comprenant les données entrantes, sortantes, et transférées *(C'est pour cette raison qu'il faut avoir un accès direct au serveur et ne pas y être connecté en SSH)*

iptables -t filter -P INPUT DROP

iptables -t filter -P FORWARD DROP

iptables -t filter -P OUTPUT DROP

Nous laissons ouvertes les connexions déjà établies :

iptables -A INPUT -m state –state RELATED ETABLISHED -j ACCEPT

iptables -A OUTPUT -m state –state RELATED ETABLISHED -j ACCEPT

Enfin, nous autorisons le **loopback** afin de ne pas se bloquer soi-même *(C'est à dire le **localhost**)*:

iptables -t filter -A INPUT -i lo -j ACCEPT

iptables -t filter -A OUTPUT -i lo -j ACCEPT

Maintenant que tout est bloqué, nous pouvons ouvrir un à un les ports dont nous avons besoin. Voici un exemple avec le **port 80** *(http)* :

iptables -t filter -A INPUT -p tcp –dport 80 -j ACCEPT

iptables -t filter -A OUTPUT -p tcp –dport 80 -j ACCEPT

Il vous suffit maintenant d'adapter les 2 dernières commandes aux autres ports dont vous avez besoin, comme **443** pour le **HTTPS**, le **21** si vous utilisez le **FTP**, etc… Soyez sûr d'autoriser des ports que vous utilisez réellement et non pas *"Au cas où"*, ni par habitude. Moins il y a de ports ouverts, plus votre serveur sera sécurisé.

FAÎTES UNE VEILLE RÉGULIÈRE

SUR L'ACTUALITÉ DE WORDPRESS

Si vous avez suivi le guide jusqu'ici, vous devriez avoir écarté les principales vulnérabilités qui se trouvaient sur votre site. Mais cependant, il faut garder en tête qu'en informatique, rien n'est invulnérable, et surtout que les choses évoluent très rapidement, pour les choses positives comme pour les choses négatives.. Il faut donc s'informer régulièrement sur les nouvelles failles découvertes afin de pouvoir y trouver une solution **AVANT** de subir une attaque. Vous pouvez régulièrement consulter des sites ou des profils sur les réseaux sociaux spécialisés en sécurité informatique *(Hashtags **#vulnerability**, **#vuln**, **#infosec**, **#security**, **#databreach**)*. Selon votre activité et vos besoins, vous pouvez vous informer une fois à plusieurs fois par semaine, voire même une fois par jour si vous en avez la possibilité.

Pour ce faire, voici une liste de liens à consulter :

- https://wpvulndb.com/
- https://www.exploit-db.com/browse/
- https://www.cvedetails.com/vulnerability-list/vendor_id-2337/product_id-4096/
- https://www.rapid7.com/db/search?q=wordpress&t=a
- https://twitter.com/search?q=%23wordpress%20%23security&src=typd

Faîtes régulièrement des backups

Afin de prévoir tout dysfonctionnement *(Qu'il vienne d'un pirate, d'un simple bug ou de vous)*, il est fortement conseillé de faire régulièrement des backups, c'est à dire une sauvegarde complète de votre site ou de votre blog. Les administrateurs systèmes disposent d'un grand nombre méthodes pour y parvenir, qu'elles soient manuelles, semi-automatisées ou automatisées, ainsi que des replica *(Une sauvegarde de la sauvegarde sur un autre serveur)*. Mais pour le commun des mortels, il existe des plugins qui permettent de simplifier cette tâche. En voici une liste non exhaustive. À vous de les comparer et de voir lequel vous correspond le plus, en fonction de votre matériel, des services que vous utilisez *(Mail, cloud, serveurs, NAS, etc…)* ainsi que des connaissances techniques que vous avez *(Certains plugins sont plus accessibles que d'autres techniquement parlant)*.

- https://wordpress.org/plugins/wp-db-backup/
- https://wordpress.org/plugins/backupwordpress/
- https://wordpress.org/plugins/backwpup/
- https://wordpress.org/plugins/updraftplus/
- https://wordpress.org/plugins/client-dash-backup-buddy/
- https://wordpress.org/plugins/duplicator/
- https://wordpress.org/plugins/vaultpress/

Bannissez le FTP au profit du SFTP

Plutôt que de répéter ce que vous pouvez trouver sur d'autres sites, je vais juste résumer la différence entre FTP et SFTP avant de vous donner 2 liens qui pourront vous permettre d'approfondir le sujet, si vous le souhaitez.

Pour faire court, un protocole FTP est un protocole qui permet l'échange de fichiers entre plusieurs machines *(PC, serveur, NAS, RaspberryPi, etc…)*. Il faut surtout retenir que le SFTP est un FTP mieux sécurisé *(Encore une fois, rien n'est invulnérable)*. Certains hébergeurs proposent les deux, d'autres n'en proposent qu'un seul. Si vous en avez la possibilité, choisissez le SFTP. Si vous avez vos propres serveurs vous avez la liberté du choix, mais si tel était le cas, vous ne seriez probablement pas en train de lire ce guide.

Voici quelques précisions pour ceux qui sont intéressés :

Explication courte :
http://www.vb-factoring-bank.at/hod/fr/help/ftpoverview.html

Explication plus détaillée :
https://openclassrooms.com/courses/creer-son-serveur-ftp

Privilégiez le serveur dédié ou le VPS au serveur mutualisé

Les hébergeurs proposent plusieurs types de serveurs. Principalement les serveurs dédiés, les serveurs mutualisés et les VPS. Voyons en quelques mots la différence entre ces 3 solutions.

Pour résumer, avec un serveur dédié, vous disposez d'un serveur complet, il est pour vous. Vous pouvez ainsi y installer le système d'exploitation que vous désirez et l'administrer de la manière que vous préférez, apportant sont lot d'avantages mais aussi les éventuelles erreurs que vous pourriez commettre. C'est un peu comme votre propre serveur que vous auriez chez vous, sauf que vous n'avez pas à gérer l'évolution du matériel et les éventuelles pannes, ces dernières étant traitées par l'hébergeur en toute transparence, ou presque… *#OVHGATE*

Un serveur mutualisé apporte les mêmes avantages qu'un serveur dédié, à la seule différence que vous le partagez avec d'autres sites web *(Donc d'autres personnes et/ou entreprises)*. De fait, les prix sont moins chers, mais de nouveaux inconvénients apparaissent. Le premier étant que vous avez moins de liberté quant au choix du système d'exploitation, mais ce n'est pas dramatique. Un autre inconvénient, plus important : si, sur le serveur où se trouve votre site, un des autres sites web hébergé est vulnérable, un cracker peut alors y accéder, s'octroyer les droits

admin du serveur et ainsi accéder aux autres sites *(Le votre inclus)*. Si les serveurs mutualisés dont de mieux en mieux sécurisés, il faut garder en tête que les crackers sont aussi de mieux en mieux armés et qu'un site mal géré peut affaiblir celui des autres personnes hébergées à côté. Dernier inconvénient majeur, les ressources du serveur *(processeur, mémoire, etc...)* sont aussi partagées entre les différents sites qu'il héberge. Si un des sites du serveur a une certaine notoriété, accueillant plusieurs milliers ou dizaines de milliers de visiteurs par heure, il peut alors être difficile d'avoir une bande passante correcte pour les autres sites qui deviendront lents et difficiles à parcourir pour les visiteurs *(Mais c'est de plus en plus rare de nos jours)*.

Enfin, le VPS est un mix des serveurs dédiés et mutualisés. Plutôt que d'avoir un serveur physique divisé en plusieurs parties, comme dans le cas d'un serveur dédié, il le sépare grâce un système de virtualisation, beaucoup plus sécurisé *(Même si Meltdown et Spectre changent la donne)*. Il offre aussi plus de flexibilité à l'utilisateur qui peut administrer quasiment comme un serveur dédié, nottament pour le choix du système d'exploitation, mais aussi en termes de performances ou de sécurité, entre autres. Vous obtenez ainsi les avantages des deux types de serveurs vus précédemment.

Voici quelques précisions pour aller plus loin :

- https://www.ivision.fr/hebergement-mutualise-dedie-vps-quelles-differences/
- https://fr.wikipedia.org/wiki/H%C3%A9bergement_mutualis%C3%A9
- https://fr.wikipedia.org/wiki/Serveur_d%C3%A9di%C3%A9
- https://fr.wikipedia.org/wiki/Serveur_d%C3%A9di%C3%A9_virtuel

Conclusion

Nous voici maintenant arrivé à la fin de ce guide. J’espère que vous y avez appris des choses et qu’elles vous seront utiles, à l’avenir, afin de renforcer la sécurité de votre site web. Les intrusions sont toujours une étape désagréable à vivre, au même titre que le cambriolage de votre logement. Elles peuvent engendrer des pertes de données, voire même une modification de votre contenu, de votre image de marque, allant parfois jusqu’à la fermeture de l’entreprise touchée.

Une fois que vous aurez corrigé vos mauvaises habitudes et fait du tri dans les plugins grâce aux premiers chapitres, pensez bien à revenir régulièrement dans la partie “Bonnes pratiques” afin de maintenir le travail que vous aurez effectués. Mieux vaut passer 10 ou 20 minutes par semaine pour maintenir la sécurité de votre site plutôt que de payer un développeur ou un expert en sécurité pour tout reprendre de zéro, engendrant ainsi une inactivité temporaire de votre site et un coût financier non-négligeable pour vous ou votre entreprise.

Si vous doutez de vos capacités à gérer correctement votre site sur le long terme, voici pour conclure une citation de **Joseph Finder** qui devrait vous redonner confiance en vous :

“Je ne suis pas un crack en informatique, loin de là, mais il n’y a pas que les mécaniciens qui sachent conduire une voiture”.

Glossaire

0-day : une vulnérabilité ne disposant pas de correctif connu. Les ordinateurs infectés se trouvent alors sans défense.

Administration système : un ensemble de tâches consistant à gérer les serveurs et bases de données d'un site web, d'une entreprise, ou toute autre entité utilisant des systèmes d'information.

Backup : une duplication des données d'un serveur, généralement vers un autre serveur, mais parfois sur une autre partition de ce même serveur ou sur un support externe.

Brute-force : une attaque par "force brute" permet de trouver un mot de passe en testant toutes les combinaisons possibles. Les principales techniques utilisées sont soit avec un dictionnaire de mots de passe que le script parcours un à un, soit en testant des combinaisons de caractères alphanumériques et spéciaux de différentes longueurs jusqu'à ce que ça corresponde avec le mot de passe ciblé.

C : un langage de programmation écrit en 1972. Il est généraliste et est à l'origine de nombreux logiciels et systèmes d'exploitation. Il a inspiré d'autres langages plus modernes comme le C++, le C# de Microsoft ou l'Objective C d'Apple.

CMS (Content Management System, ou en français Système de Gestion de Contenu) : un site web prêt à l'emploi comprenant une partie Administrateur déjà conçue, et une partie client *(Celle visible par l'internaute qui visite le site)* déjà pré-faite mais que vous pourrez modifier à votre guise à l'aide de thèmes, extensions ou en modifiant directement le code source. Il a l'avantage d'être accessible aux plus

débutants mais aussi d'être flexible pour les personnes ayant des connaissances plus ou moins approfondies en code.

Cookies: De petits fichiers texte qui stockent les informations à mémoriser d'un site *(Nom d'utilisateur, mot de passe, changement de l'apparence, abonnement Premium, inscrit ou visiteur, contenu du panier lors de la dernière visite, etc…).* C'est grâce aux cookies que vous pouvez vous reconnecter de manière transparente sur un site que vous consultez régulièrement, que la couleur de thème que vous aviez sélectionné réapparait automatiquement, ou que l'aspirateur que vous aviez ajouté au panier y apparait toujours à chaque nouvelle visite.

Hydra : un logiciel écrit en **C*** permettant de craquer des mots de passe en utilisant diverses tecnhniques.

.htaccess : un fichier de configuration pour les serveurs Apache principalement utilisé pour définir des droits d'accès, mais aussi pour faire des redirections ou pour afficher des messages d'erreurs.

Injection SQL : un ensemble de techniques permettant d'exploiter les failles des bases de données permettant d'exécuter des commandes avec les droits aministrateurs ou de récupérer les données des utilisateurs, entres autres.

Meltdown et Spectre : Un ensemble de failles (Une pour Meltdown et deux pour Spectre) annoncées le 2 janvier 2018. Les vulnérabilités les plus courantes sont souvent logicielles, au niveau des programmes ou du système d'exploitation. Ici, ces failles sont matérielles et touchent directement les CPU *(les processeurs)*, ceux construits entre 1995 et 2018. Les Intel sont les plus touchés, mais d'autres architectures comme AMD et ARM le sont aussi, ce qui signifie que tous les types de matériels sont touchés, du simple PC au serveur en passant par le Raspberry Pi ou la

montre connectée, sans oublier le très démocratisé smartphone. Meltdown affaiblit l'isolation entre le kernel *(Le noyau du système d'exploitation)* et les applications utilisées, alors que Spectre, lui, affaiblit l'isolation entre les applications elles-mêmes. Si par exemple votre navigateur web était corrompu, un cracker pourrait accéder à votre gestionnaire de mots de passe ou à votre presse-papier quand vous faîtes un copier/coller. Les principaux acteurs de l'informatique travaillent d'arrache-pied pour minimiser les dégâts, mais ne permettront probablement jamais de les patcher à 100%. Les seules solutions vraiment fiables étant de changer le processeur de sa machine ou d'acheter un nouveau PC avec de nouveaux processeurs quand les consructeurs qui seront fabriqués courant 2018.

MySQL : un système, partiellement libre, permettant de gérer des bases de données. C'est le SGBD *(Système de Gestion de Base de Données)* le plus utilisé au monde.

Pare-feu : un programme permettant d'analyser les données entrantes et sortantes sur un ordinateur ou un serveur, permettant ensuite d'interdire certains accès si quelque chose semble anormal ou malveillant.

PHP : un langage de programmation libre, créé en 1994, permettant de créer des sites web dynamiques. Il permet à une page de s'adapter à chaque profil utilisateur *(En affichant les données de son compte, comme sur les réseaux sociaux ou les boîtes email par exemple)*, à chaque pays *(Choix automatique de la langue en fonction de la géolocalisation)*, etc… Sans ce type de langage, on parle alors de site web statique *(L'apparence et le contenu resteront toujours les mêmes, quoi qu'il arrive)*.

PhpMyAdmin : une application web écrite en PHP permettant de gérer graphiquement, dans un navigateur web, des bases de données.

Ruby : un langage de programmation libre créé au Japon en 1995

Serveur : techniquement proche de l'ordinateur que vous avez chez vous mais avec des performances différentes, un serveur permet de mettre à disposition un ensemble de services très divers *(Site web, boîte email, compte bancaire, etc...)* pour des clients *(Les logiciels utilisés par les internautes)*. Dans le cas de notre site sous Wordpress, le site serait hébergé *(Stocké et mis à disposition)* sur un serveur afin qu'un internaute *(Le client)* puisse le consulter via son navigateur.

Web 2.0 : terme désignant le Word Wide Web *(C'est de là que viennent les "www" d'une adresse web)* dans une version simplifiée grâce à de nombreuses innovations et à une conception plus collective, ceci afin d'être accessible au plus grand nombre. C'est cette simplicité apparue au début des années 2000 qui lui a permise d'être aussi démocratisé de nos jours.

WPScan : un script écrit en **Ruby*** permettant d'analyser un site sous Wordpress et de tester sa sécurité.

XSS (Cross Shell Scripting) : un type d'attaque permettant d'ajouer du contenu sur une page d'un site web, afin d'en changer son apparence et/ou de lui faire exécuter des tâches non-prévues par les concepteurs du site.

Écrit par **Franck Ridel** à **Bucarest**

Écriture achevée en **janvier 2018**

https://franck-ridel.fr

Cet ebook est publié sous licence Creative Commons **CC BY-ND**